```
rel_star_of_equiv  THM

⊢  ∀T:Type, E:(T→T→Prop), x,y:T.
|  EquivRel(T)(_1 E _2) ⇒ (x (E^*) y) ⇒ (x E y)
| by Unfold ‘rel_star‘ 0 THEN Reduce 0 THEN Auto THEN ExRepD
|
1. T : Type
2. E : T→T→Prop
3. x : T
4. y : T
5. EquivRel(T)(_1 E _2)
6. n : ℕ
7. x E^n y
⊢  x E y
| by MoveToConcl (-1) THEN MoveToConcl 4 THEN MoveToConcl 3 THEN

|    MoveToConcl (-1) THEN InductionOnNat
|
|\
| .....basecase .....
| 1. T : Type
| 2. E : T→T→Prop
| 3. EquivRel(T)(_1 E _2)
| ⊢  ∀x,y:T. (x E^0 y) ⇒ (x E y)
| | by RecUnfold ‘rel_exp‘ 0 THEN Reduce 0 THEN Auto
| |
| 1. T : Type
| 2. E : T→T→Prop
| 3. EquivRel(T)(_1 E _2)
| 4. x : T
| 5. y : T
| 6. x = y
| ⊢  x E y  by HypSubst (-1) 0 THENA Auto THEN UseEquiv
|
.\
. .....upcase .....
. 1. T : Type
. 2. E : T→T→Prop
. 3. EquivRel(T)(_1 E _2)
. 4. n : ℤ
. 5. 0<n
. 6. ∀x,y:T. (x E^n-1 y) ⇒ (x E y)
. ⊢  ∀x,y:T. (x E^n y) ⇒ (x E y)
. | by ...
. |
. |\
. | .....truecase .....
. | 1. T : Type
. | 2. E : T→T→Prop
. | 3. EquivRel(T)(_1 E _2)
. | 4. n : ℤ
. | 5. 0<n
. | 6. ∀x,y:T. (x E^n-1 y) ⇒ (x E y)
. | 7. n = 0
. | ⊢  ∀x@0,y:T. (x@0 (λx,y. x = y) y) ⇒ (x@0 E y)  by Auto
```

```
. |
. .\
. . .....falsecase .....
. . 1. T : Type
. . 2. E : T→T→Prop
. . 3. EquivRel(T)(_1 E _2)
. . 4. n : ℤ
. . 5. 0<n
. . 6. ∀x,y:T. (x E^n-1 y) ⇒ (x E y)
. . 7. ¬n = 0
. . ⊢  ∀x,y@0:T.
. . |  (x (λx,y. ∃z:T. (x E z) & (z E^n-1 y)) y@0) ⇒ (x E y@0)
. . | by Reduce 0 THEN Auto
. . |
. . 1. T : Type
. . 2. E : T→T→Prop
. . 3. EquivRel(T)(_1 E _2)
. . 4. n : ℤ
. . 5. 0<n
. . 6. ∀x,y:T. (x E^n-1 y) ⇒ (x E y)
. . 7. ¬n = 0
. . 8. x : T
. . 9. y@0 : T
. . 10. ∃z:T. (x E z) & (z E^n-1 y@0)
. . ⊢  x E y@0  by ExRepD THEN UseTrans z THEN Auto
. . |
. . 1. T : Type
. . 2. E : T→T→Prop
. . 3. Refl(T)(_1 E _2)
. . 4. Sym _1,_2:T. _1 E _2
. . 5. Trans x,y:T. x E y
. . 6. n : ℤ
. . 7. 0<n
. . 8. ∀x,y:T. (x E^n-1 y) ⇒ (x E y)
. . 9. ¬n = 0
. . 10. x : T
. . 11. y@0 : T
. . 12. z : T
. . 13. x E z
. . 14. z E^n-1 y@0
. . ⊢  z E y@0  by BackThruSomeHyp THEN Auto'
END
```